

FIGURE 1
(PRIOR ART)

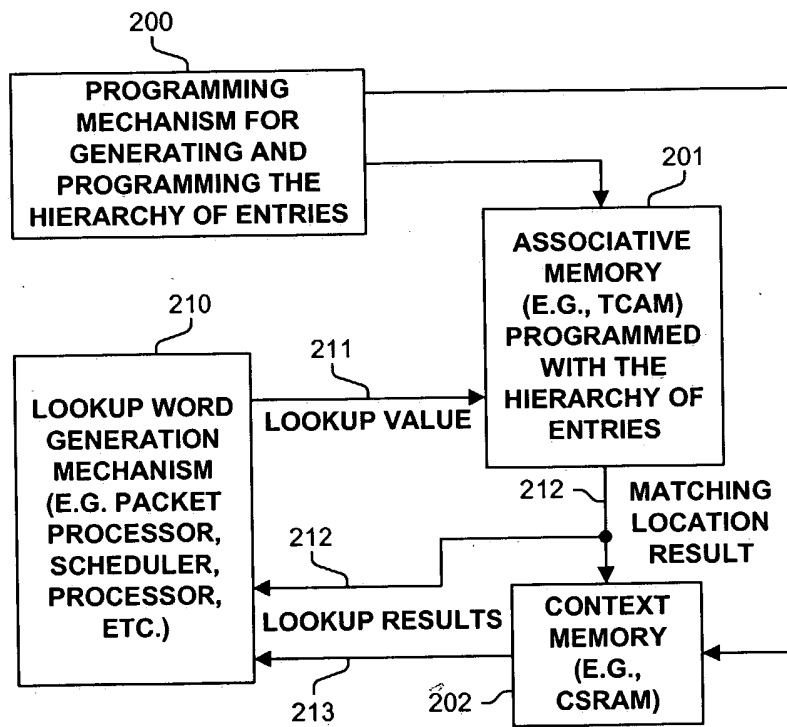


FIGURE 2A

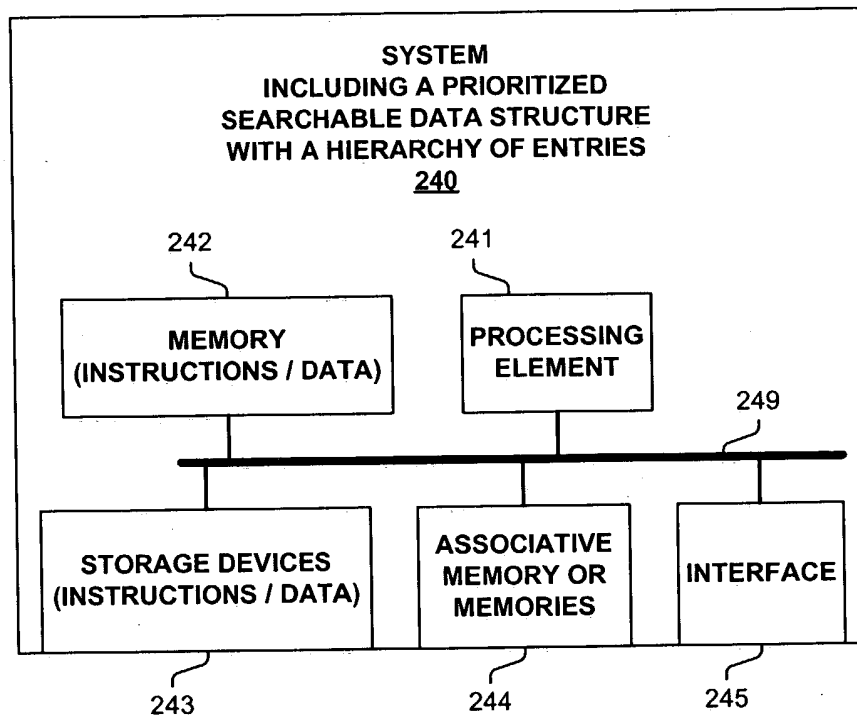


FIGURE 2B

**PRIORITIZED
SEARCHABLE DATA
STRUCTURE
(E.G., ASSOCIATIVE
MEMORY ENTRIES)
300**

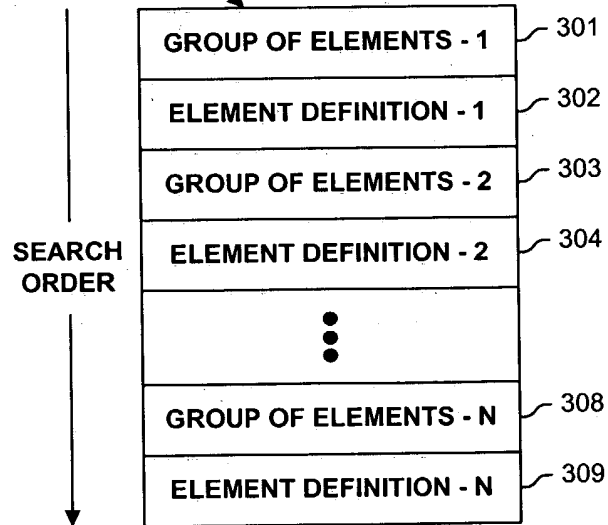


FIGURE 3A

**PRIORITIZED SEARCHABLE
DATA STRUCTURE
(E.G., ASSOCIATIVE MEMORY
ENTRIES)
310**

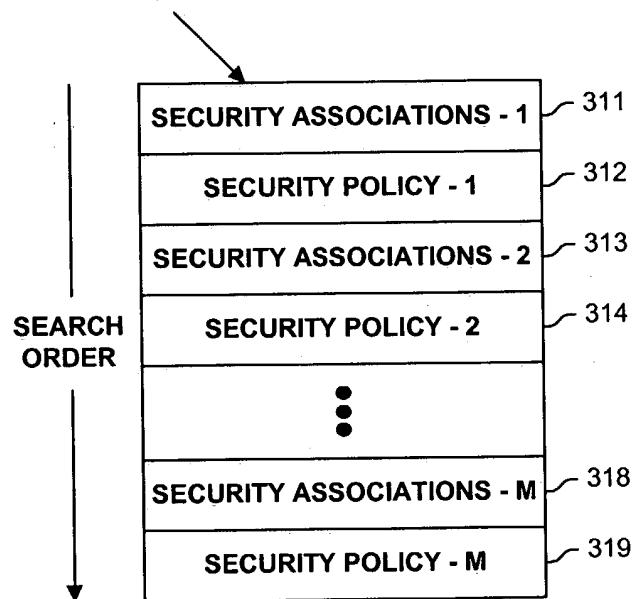


FIGURE 3B

PRIORITIZED
SEARCHABLE DATA
STRUCTURE
(E.G., ASSOCIATIVE
MEMORY ENTRIES)
330

SEARCH
ORDER

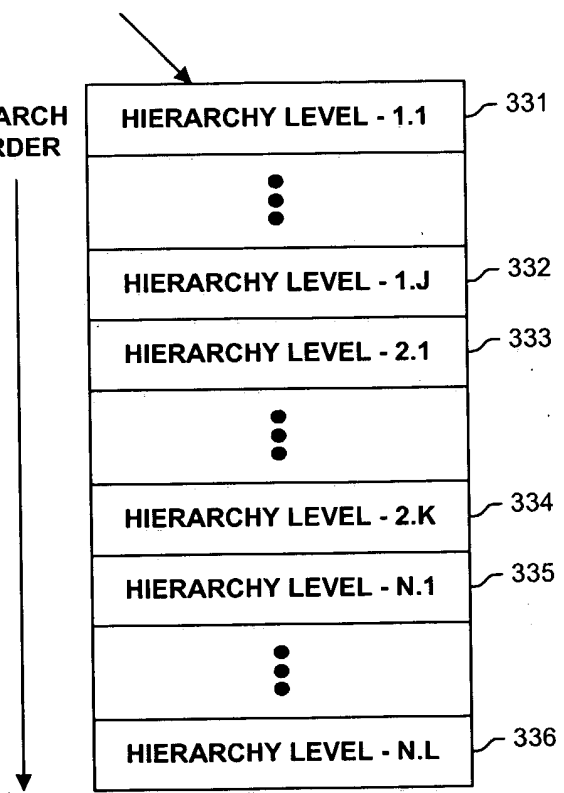
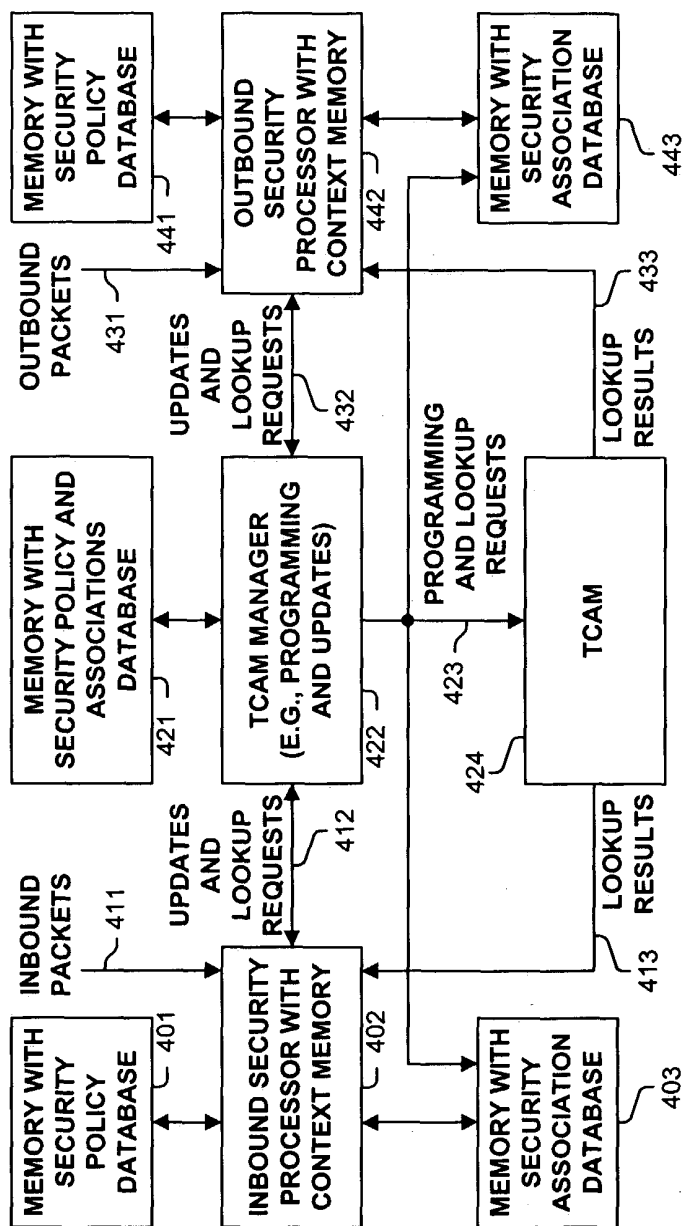


FIGURE 3C



IPSEC MECHANISM
FIGURE 4

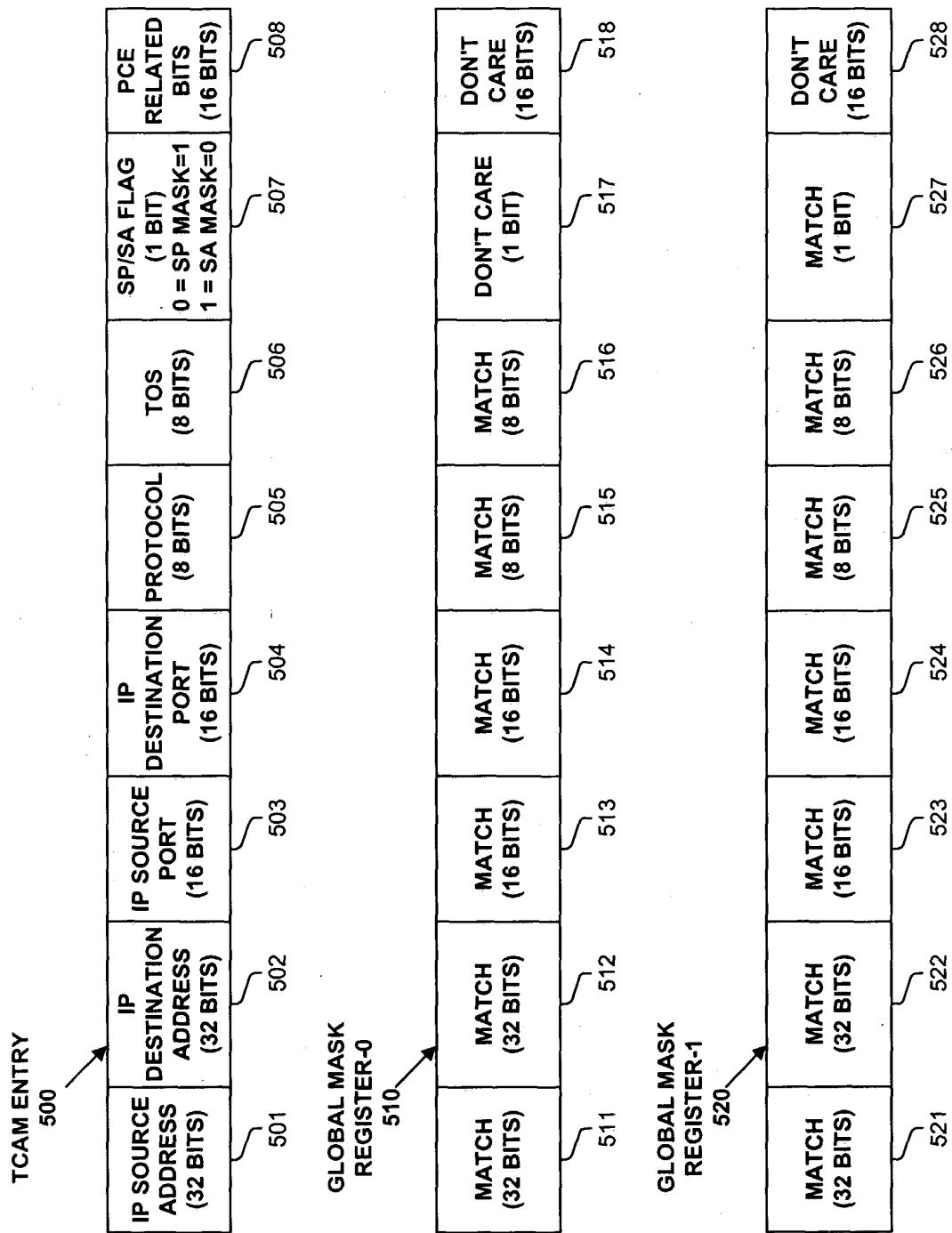


FIGURE 5A


```

Arguments: highIncl, lowIncl; // range is [highIncl, lowIncl]

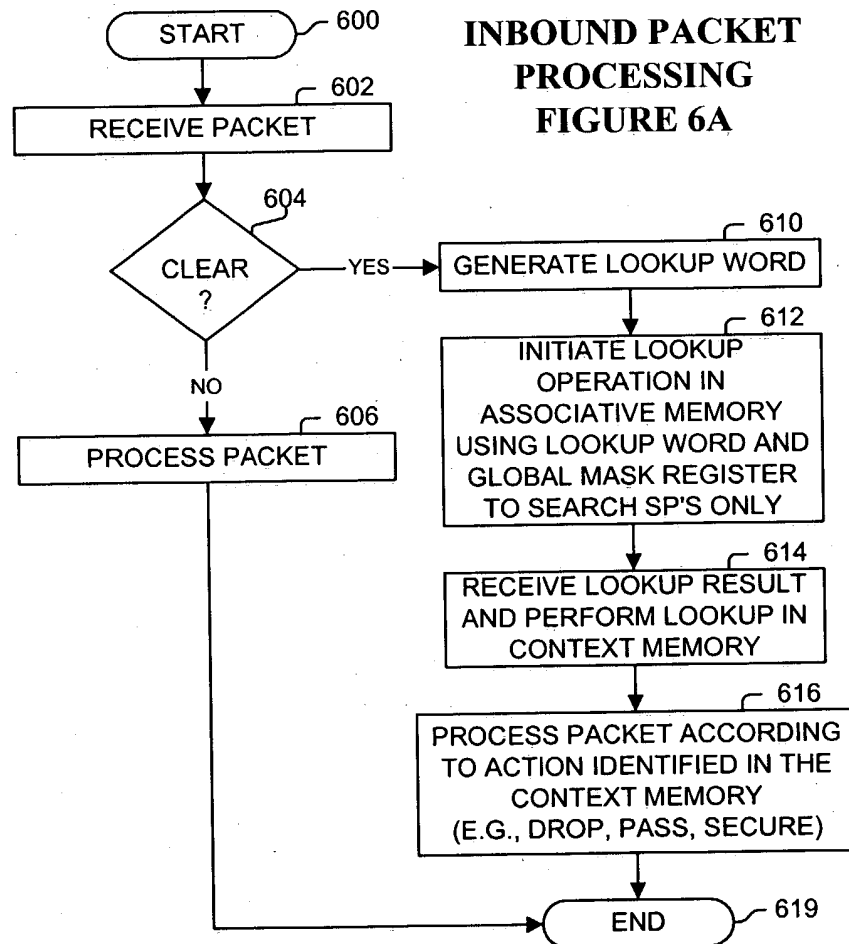
// Keep four stacks: two for results (value and masks), two for work in progress: bit and prefix
// for 16 bit fields, the results can be up to 30 TCAM entries, and the wip stacks must be 17 entries
// deep. (seems to be  $2(N-1)$  worst case entries, but I have not generated a proof!)

push (prefixStack, 0), push (bitStack, precision); // 16 for port numbers
do {
    prefix = pop (prefixStack), bit = pop (bitStack);
    mask = (1 << bit) - 1;
    highTry = prefix | mask, lowTry = prefix & ~mask;
    if (highTry <= highIncl && lowTry >= lowIncl) { // entry covers a subset of range - save it
        push (valueStack, prefix), push (maskStack, mask);
    } else if (lowTry > highIncl || highTry < lowIncl) { // entry covers a disjoint set - forget it
        // do nothing
    } else { // entry covers some value inside and outside range - split it
        push (prefixStack, prefix), push (bitStack, bit - 1);
        push (prefixStack, prefix | 1 << (bit - 1)), push (bitStack, bit - 1);
    }
} while (StackNotEmpty (prefixStack));

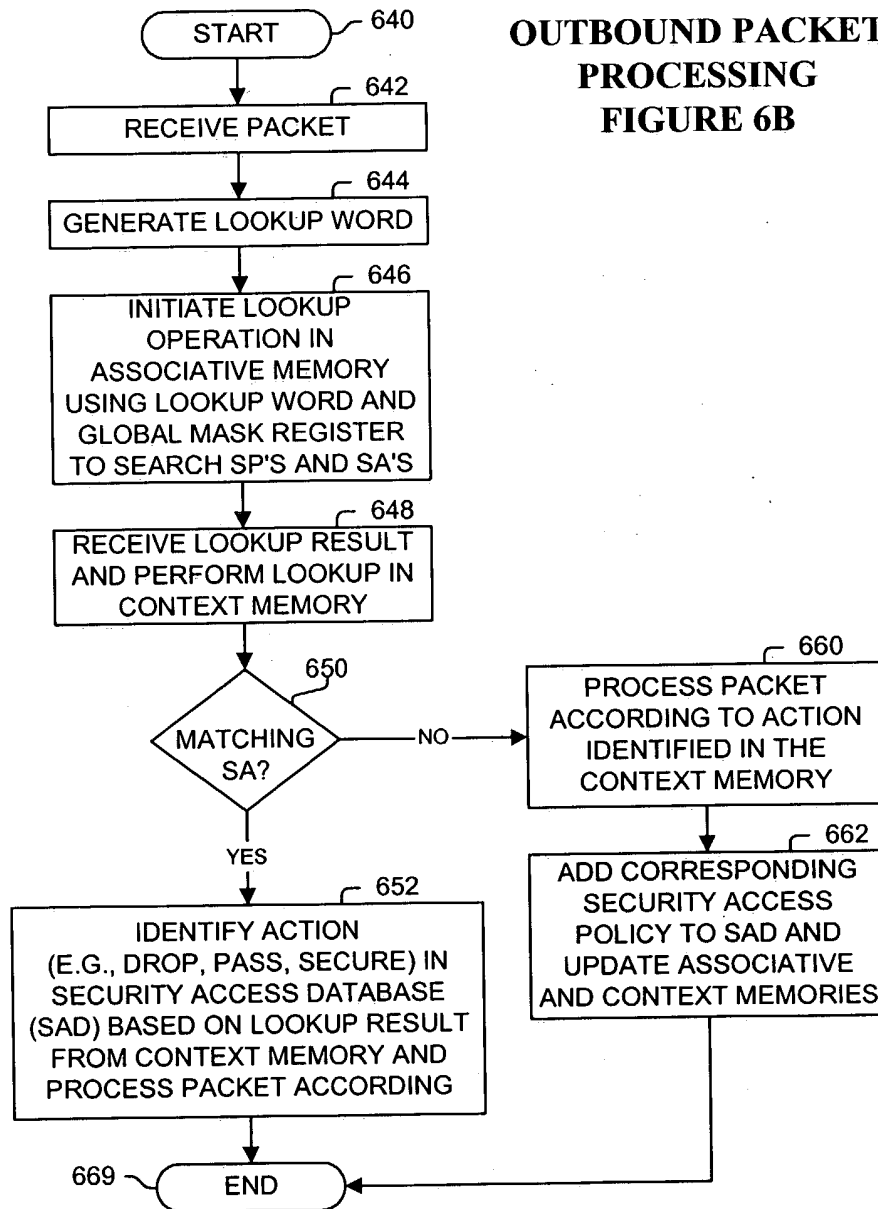
```

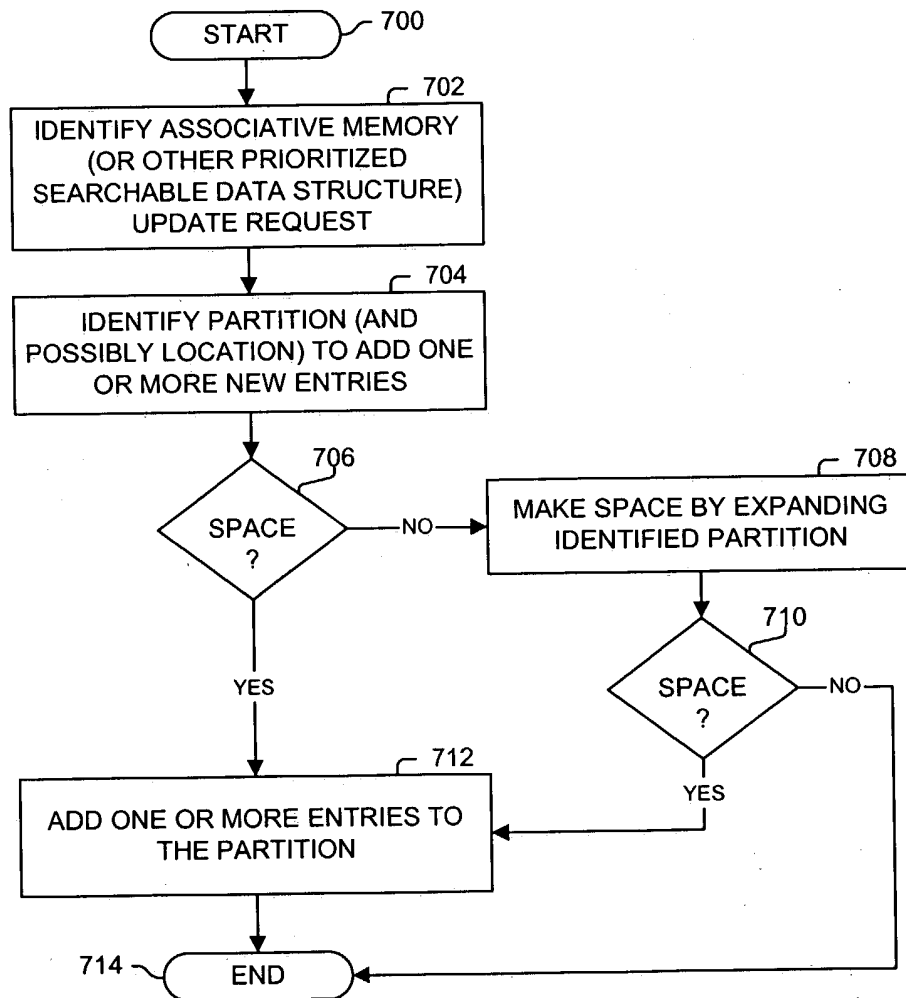
SPLITTING ENTRIES INTO MULTIPLE ENTRIES
FIGURE 5B

**INBOUND PACKET
PROCESSING
FIGURE 6A**

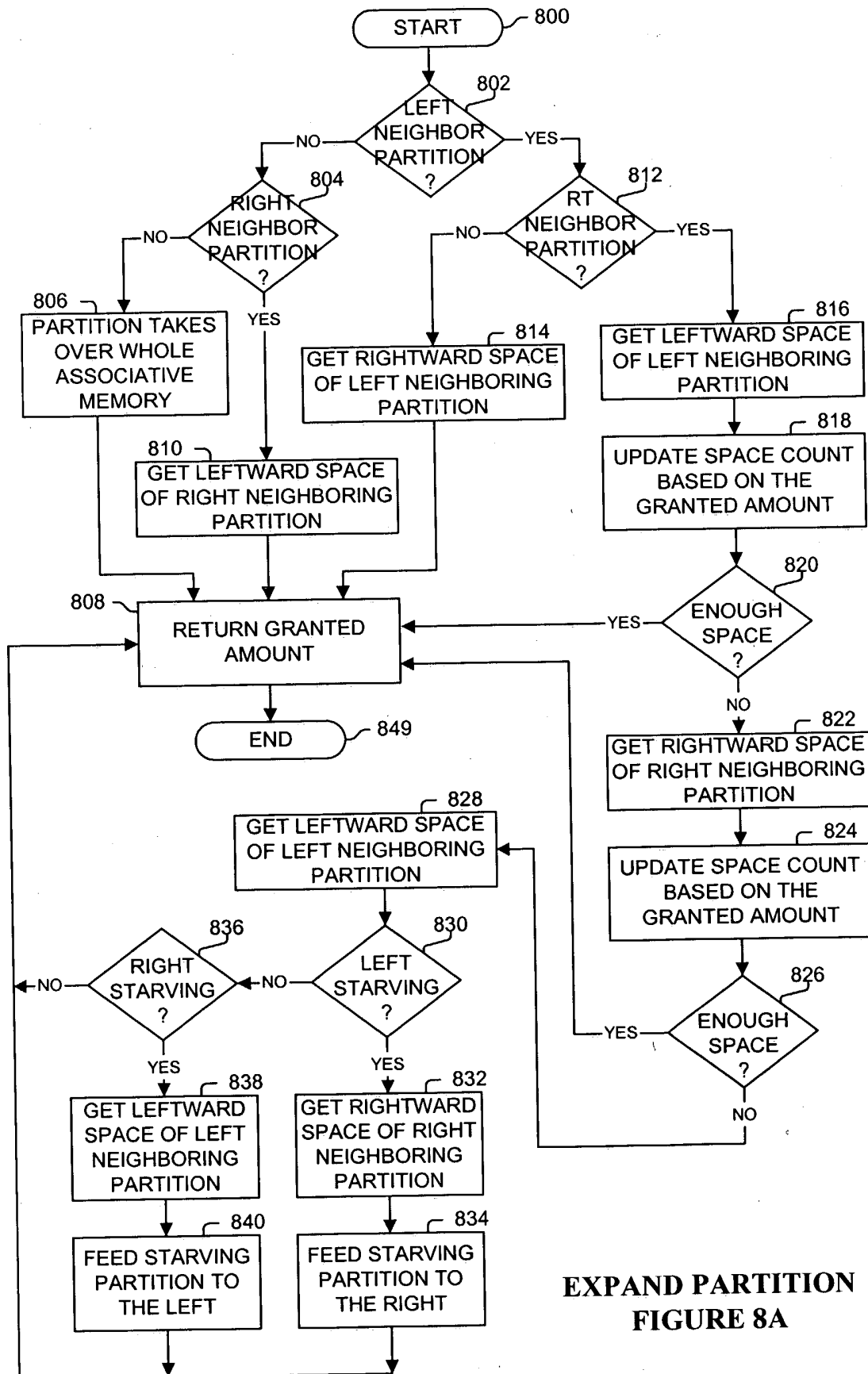


**OUTBOUND PACKET
PROCESSING
FIGURE 6B**

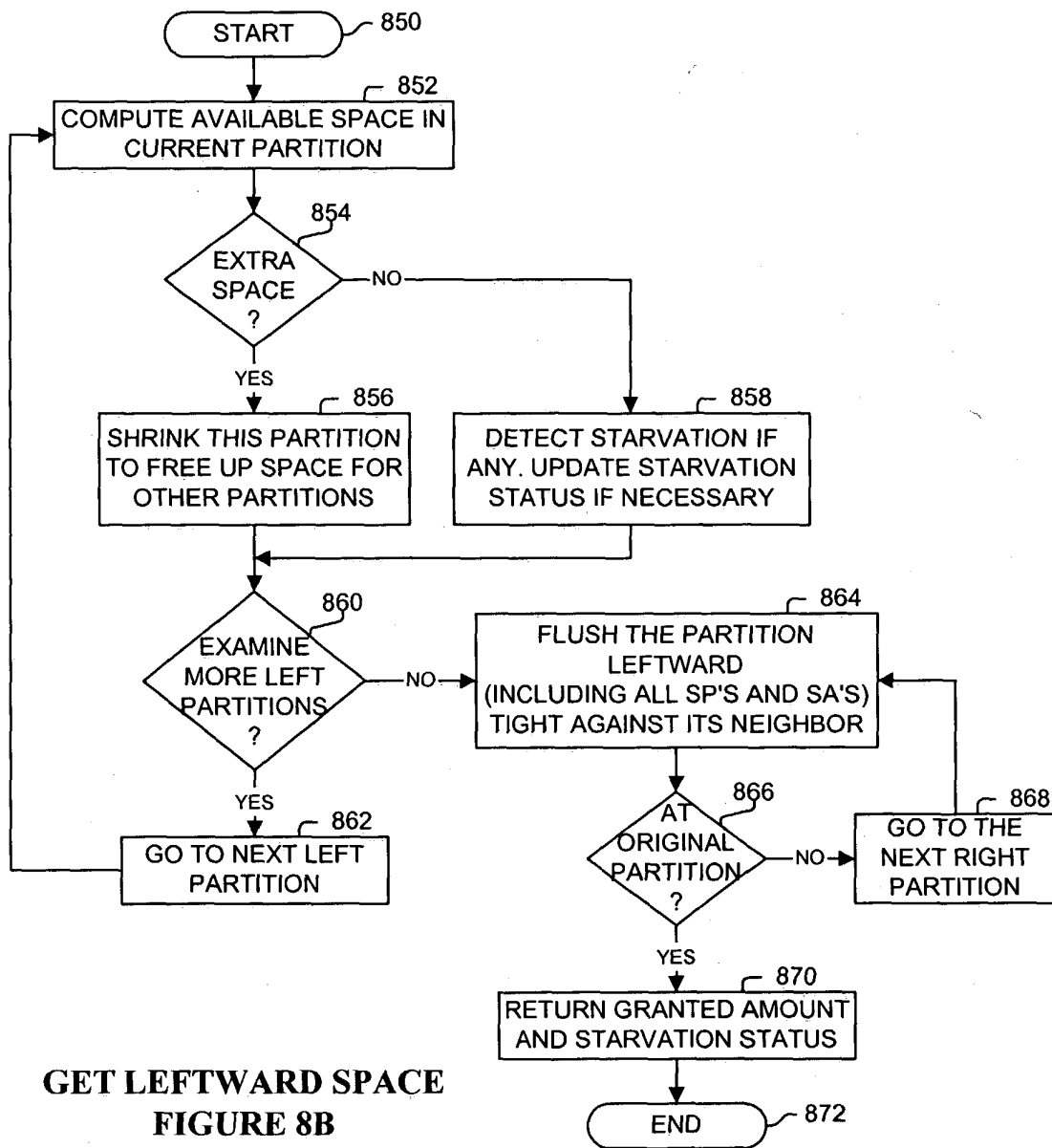


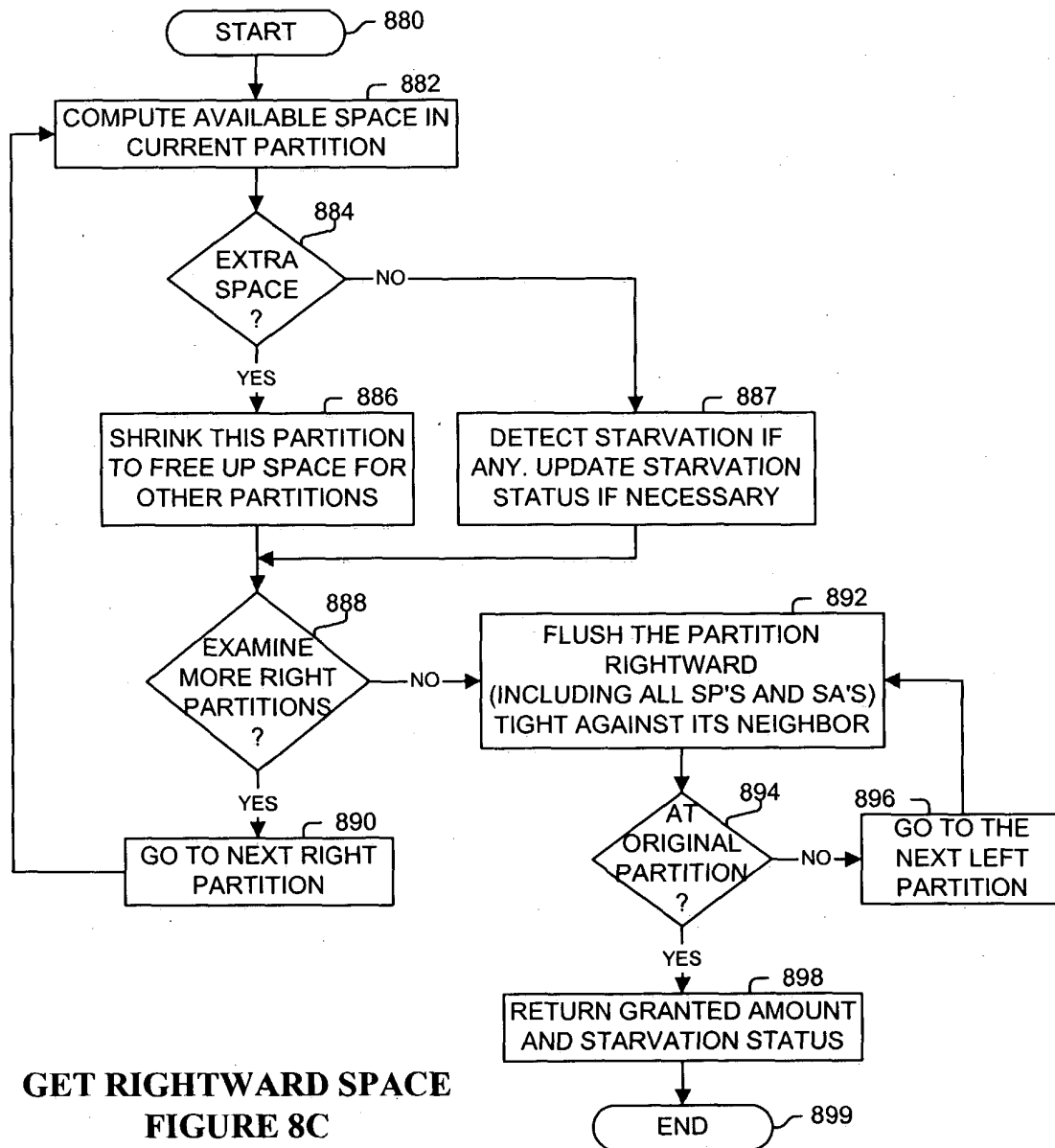


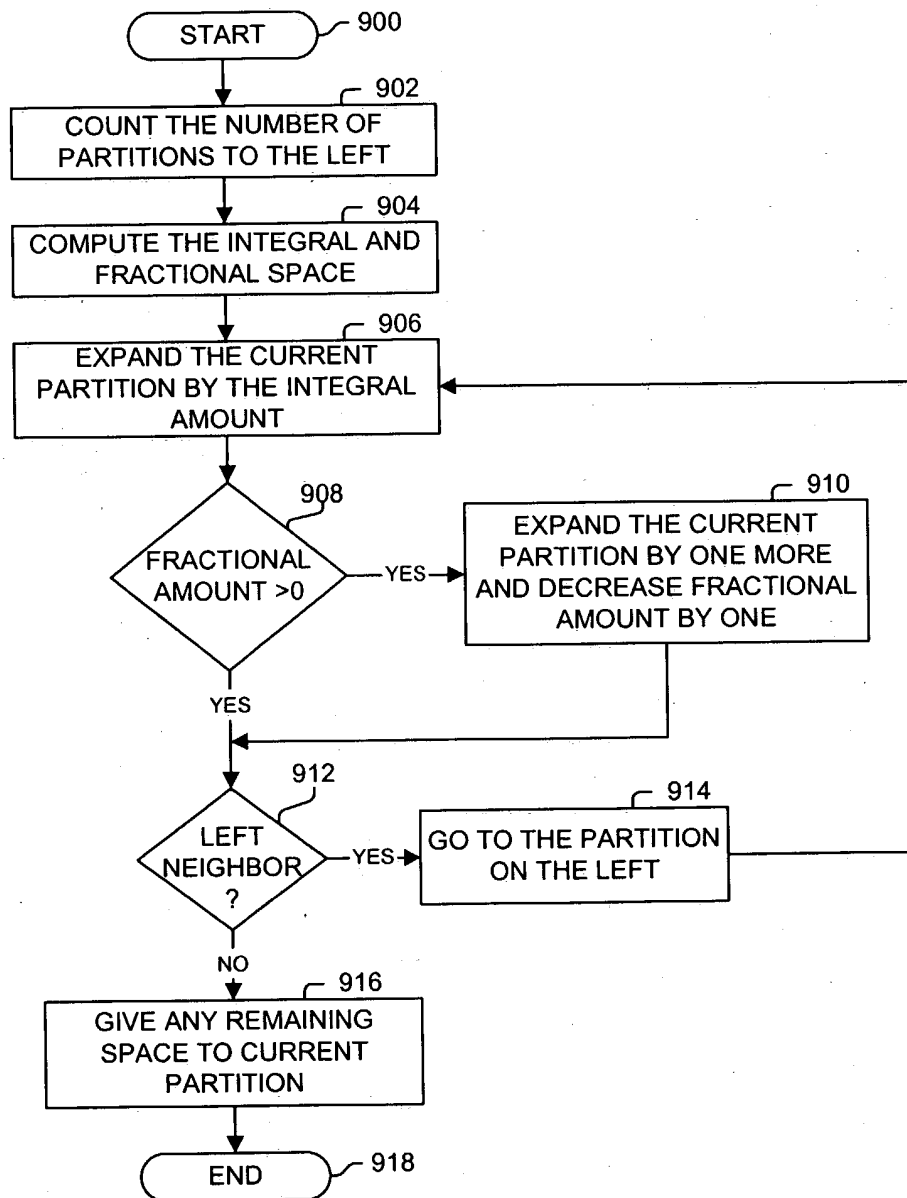
ADDING ENTRY
FIGURE 7



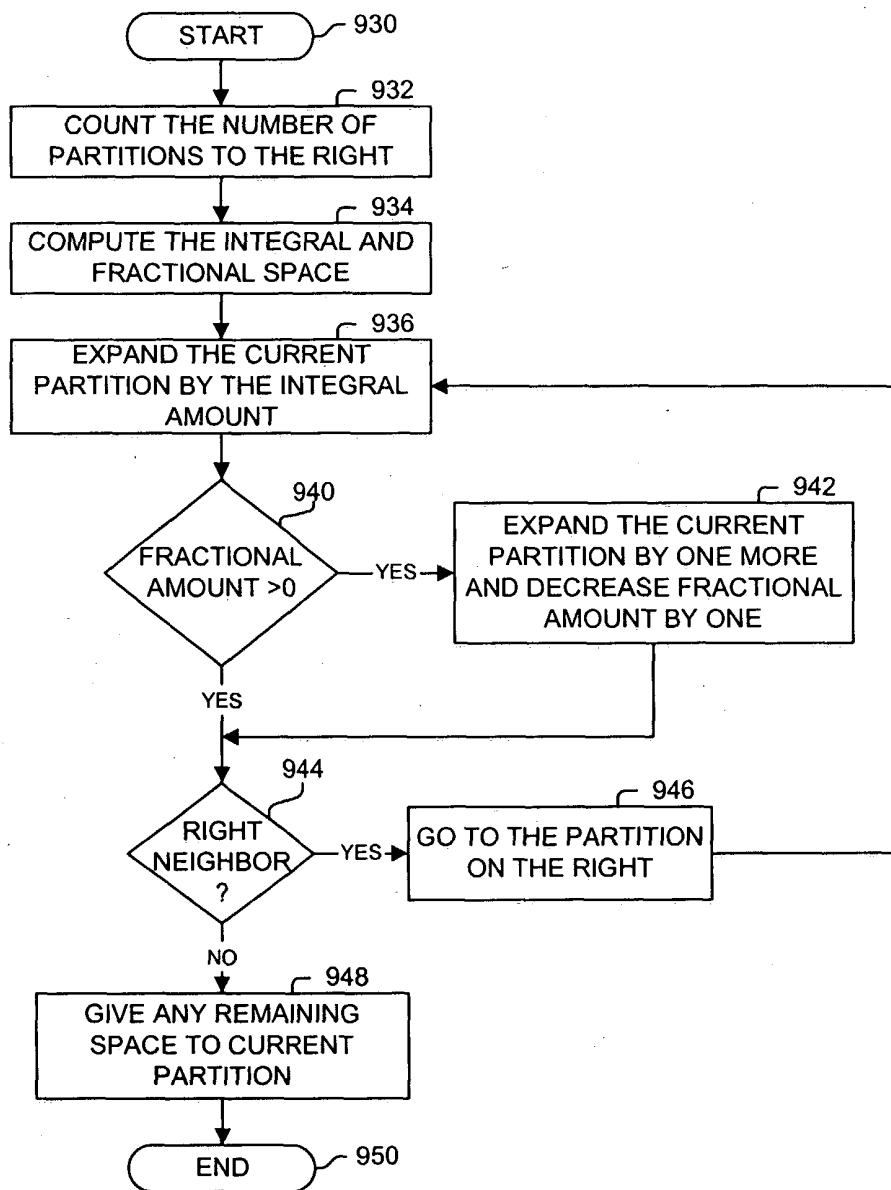
**EXPAND PARTITION
FIGURE 8A**







**FEED STARVING
LEFT PARTITION
FIGURE 9A**



**FEED STARVING
LEFT PARTITION
FIGURE 9B**